Lattice QCD in Nuclear Physics

Robert Edwards
Jefferson Lab

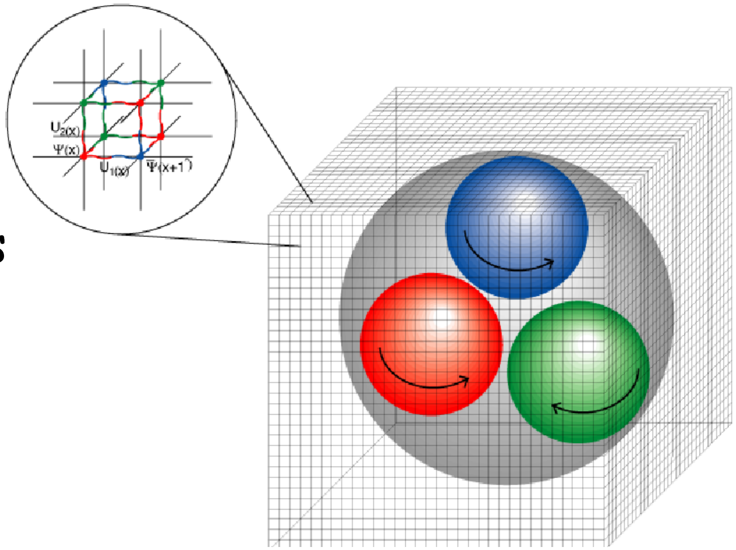
NERSC 2011

Report:
Robert Edwards, Martin Savage & Chip Watson

Current HPC Methods

- Algorithms
  - Gauge generation
  - Analysis phase
- Codes
  - USQCD SciDAC codes
  - Heavily used at NERSC: QDP++ & Chroma
- Quantities that affect the scale of the simulations
  - Lattice size, lattice spacing & pion mass

Gauge generation



Hybrid Monte Carlo (HMC)

- Hamiltonian integrator: 1st order coupled PDE's
- Large, sparse, matrix solve per step

- "Configurations" via importance sampling
- Use Metropolis method

- Produce ~1000 useful configurations in a dataset

Cost:

- Controlled by lattice size & spacing, quark mass
- Requires capability resources

Analysis

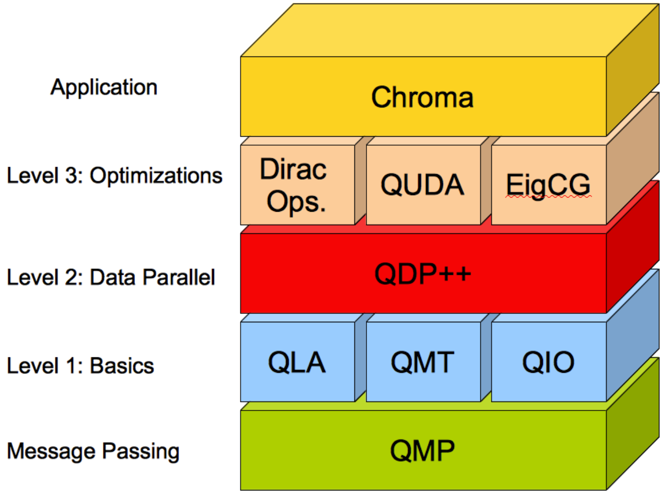
Compute observables via averages over configurations

- For one configuration:
  - Solve Dirac eqn. repeatedly: $D(A,m)\psi = \chi$
    - Large, sparse, matrix solver. Use iterative methods.
  - Either hold solutions in memory, or page to disk
  - Tie solutions together to make "observables"
- Can be 1000's of measurements for each configuration
- Repeat for 1000's of configurations

Codes

SciDAC: developed new code base – use C++

- QDP++:
  - Data parallel interface – well suited for QCD (called Level 2)
  - Hides architectural details
  - Supports threading & comms (e.g., Hybrid/MPI model)
  - Thread package: customized – can outperform OpenMP
  - Parallel file I/O support
- Chroma:
  - Built over QDP++
  - High degree of code modularity
  - Supports gauge generation
  - Task based measurement system
- Modern CS/software-engineering techniques



2010: Chroma accounted for 1/3 NERSC cycles

User base and support

- Aware of many groups using SciDAC codes
  - LHPC, NPLQCD, QCDSF, UKQCD, FNAL, etc.
- Several external applications at level 2 only
- Used extensively at sites like LLNL BG/L, clusters, GPUs, NERSC/UT/OLCF/UK Crays, ANL BG/P, TACC, QPACE (Cell)

- Software web-pages and documentation
  http://www.usqcd.org/usqcd-software
- Codes available via tarballs and Git
- Nightly builds and regression checks

- 184 citations to Chroma/QDP++ paper (Lattice 2004, Edwards & Joo')

Current HPC Requirements

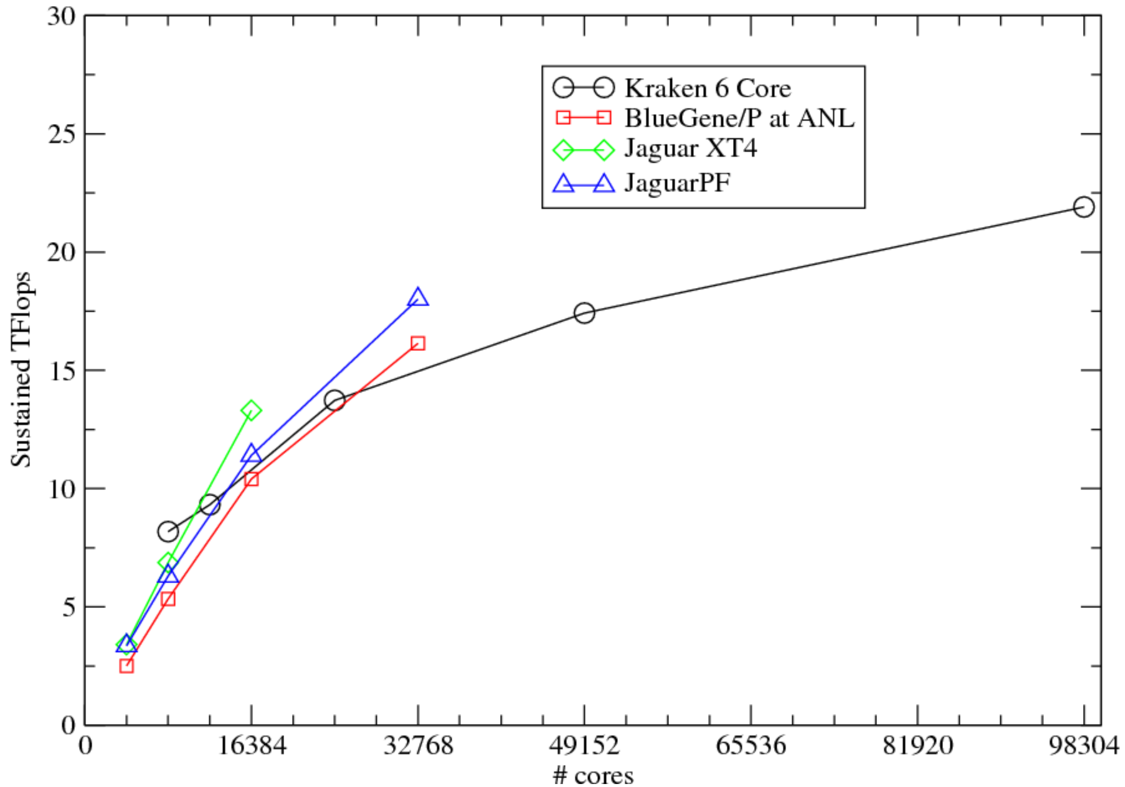
- Architectures currently used
  - Cray XT4&5 (variants), XE6, BG/L, BG/P, Linux clusters, GPUs
- Wallclock time: ~12 hours
- Compute/memory load
  - Gauge generation: 40k cores XT5 (ORNL) @ 1 GB/core
  - Analysis phase: 20k cores (NERSC XE6) @ 2 GB/core
- Data read/written
  - Gauge generation: Read/write at most 10 GB
  - Analysis phase: a) Read 10GB, write temp. ~ 1 TB, write perm 100 GB
    b) [Read 10GB, compute]@100x, write perm 100 GB
- Necessary software, services or infrastructure
  - Some analyses use Lapack
  - Analysis: global file systems useful for paging short term

Current HPC Requirements

- Hours allocated/used in 2010
  - USQCD = 58M (J/psi) + 19M BG/P + 2M (GPU) Core-Hours
    NSF Teragrid = 95 M SU's
  - Conversion Factors : XT4=XT5 =0.5 J/psi = 1.0 BG/P = SU
  - Total = 303M MPP + 2M GPU
- NERSC hours used over last year
  - 16M (charged) = 73M (used) MPP Core-Hours

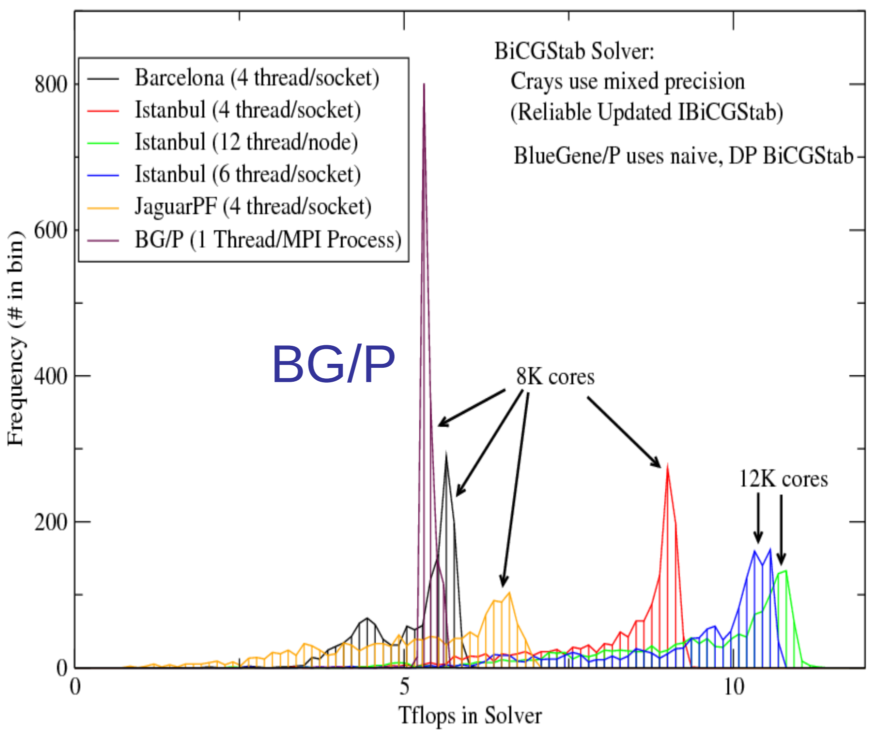
Current HPC Requirements

- Known limitations/obstacles/bottlenecks
  - Performance on Cray variants & BG/P: inverter strong scaling
  - Cray XT (<)5, hybrid MPI/threading essential

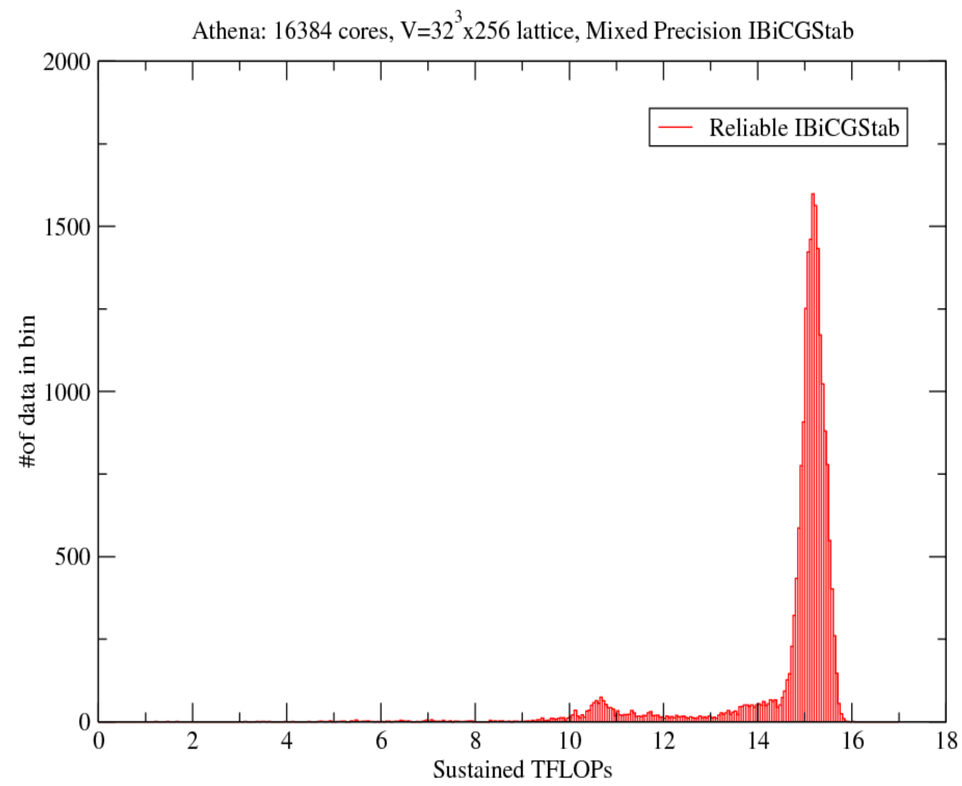


Current HPC Requirements

- Known limitations/obstacles/bottlenecks
  - Crays: loaded system -> comms interference -> perf. degradation



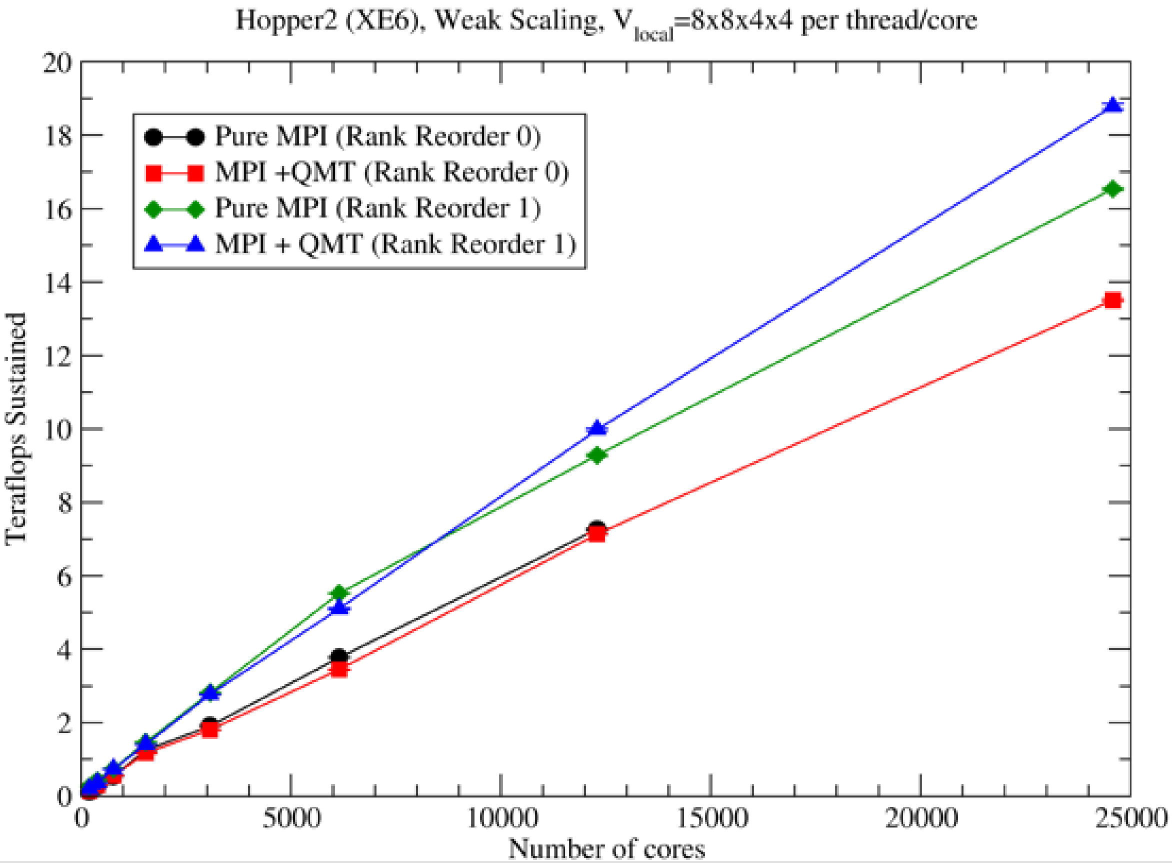
Crays (loaded)



Cray (dedicated)

Current HPC Requirements

- NERSC Cray XE6: weak scaling improved
  - hybrid threading less essential



Computational Requirements

Gauge generation : Analysis

Current calculations

- Weak matrix elements: 1 : 1
- Baryon spectroscopy: 1 : 10
- Nuclear structure: 1 : 4

Computational Requirements:

Gauge Generation : Analysis

10 : 1 (2005)

1 : 4 (2011)

Core work: Dirac inverters - use GPU-s

HPC Usage and methods for the next 3-5 years

Gauge generation:

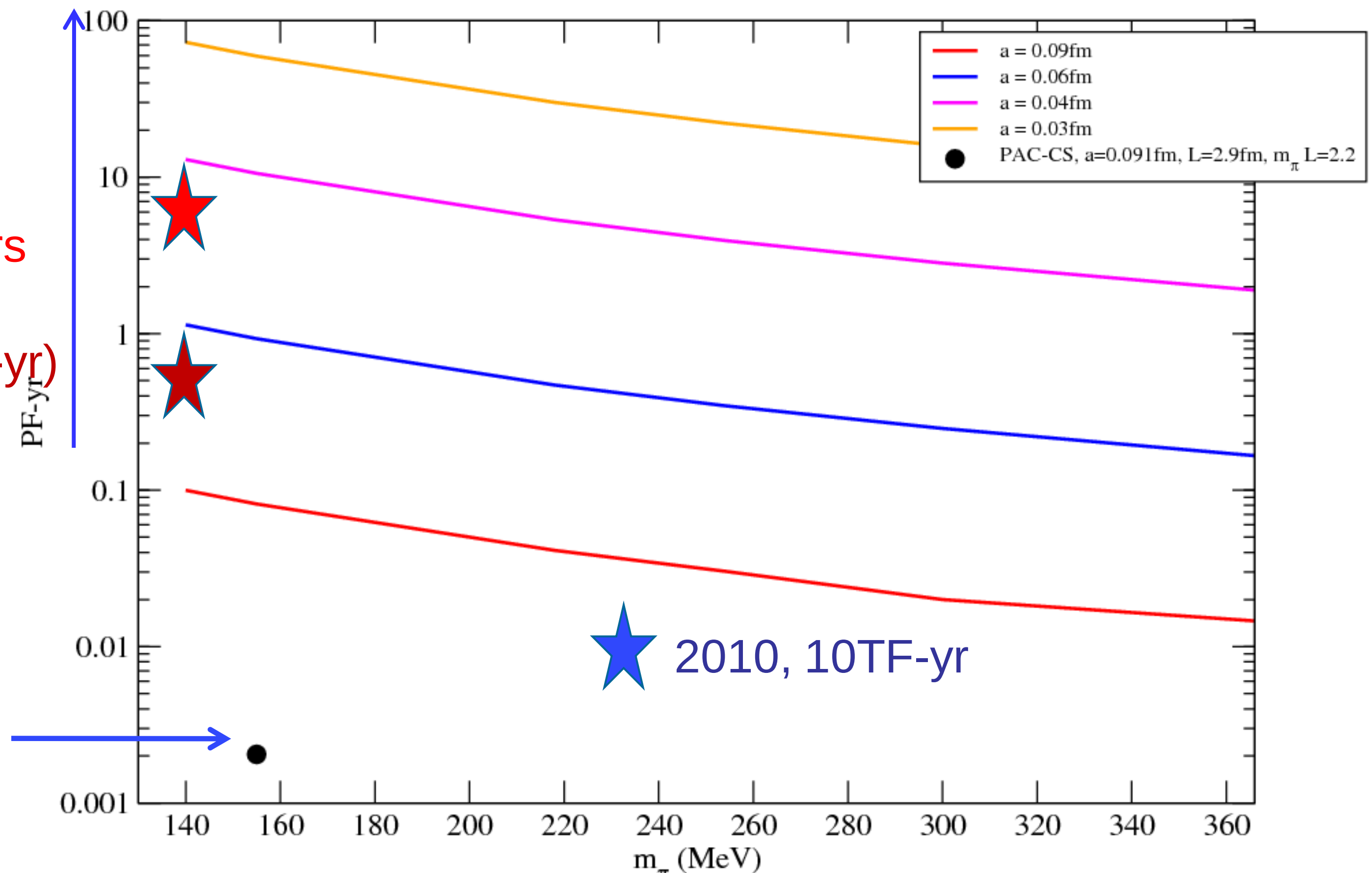
- Cost: reasonable statistics, box size and "physical" pion mass
- First milestone: 1 ensemble + analysis; second: two ensembles+analysis

PF-years

>5 PF-yr
~50 B core-hrs

2014 (0.5 PF-yr)
~4 B core-hrs

State-of-Art
2009



GPU-s boost capacity resources

- 2010 Total hours consumed: capability+capacity
  - 303M MPP + 2M GPU
- USQCD (JLab):
  - Deployed 2010: 100 TF sustained → 870 M core-hrs
  - 1 GPU ~ 100 cores
  - Ramping up: 2M GPU-hr ~ 200M core-hrs
- Capacity now increasing much faster than capability

HPC Usage and methods for the next 3-5 years

Upcoming changes to codes/methods/approaches

New algorithms & software infrastructure crucial:

- Extreme scaling: efficiency within steep memory hierarchies
- Domain decomposition techniques: new algorithms needed
- Fault tolerance – more sophisticated workflows
- Parallel IO – staging/paging
- Moving data for capacity calculations

Need SciDAC-3 and partnerships

HPC Usage and methods for the next 3-5 years

Need SciDAC-3 and partnerships

Current/future activities:

- USQCD researchers: co-designers BG/Q - improved cache usage
- Collaboration w/ Intel Parallel Computing Labs – improving codes
- NSF PRAC Bluewaters development
- Development of QUDA GPU software codes: hugely successful
- Exploiting domain decomposition techniques:
  - Push into integrators
  - Multigrid based inverters (with TOPS)
- Improving physics-level algorithms/software:
  - Measurement methods for spectroscopy, hadron & nuclear structure

Developing for Exascale

HPC Usage and methods for the next 3-5 years

- Requirements/year by 2014: 100 TF-yr = 876 M core-hrs @ 1 GF/core
- Wallclock time: still 12 to 24 hours
- Compute/memory load: typical problem: 48^3x384 (up from 32^3x256)
  - Gauge generation: typical problem: 48^3x384 (up from 32^3x256)
    - Many-core: 100k cores @ 1 GB/core
    - Heterogeneous: 1000's of gpus: e.g., 2592*4 within XK6
  - Analysis phase:
    - Many-core: 20k – 40k cores @ 2 GB/core
    - Heterogeneous: low 100's of gpus
- Data read/written
  - Gauge generation: Read/write 50 GB
  - Analysis phase: Read 50GB, write temp. 100@100GB, write perm 100 GB
- Checkpoint size: 100GB

HPC Usage and methods for the next 3-5 years

- On-line storage: 1 TB and 20 files
- Off-line storage: 100 TB and ~2000 files

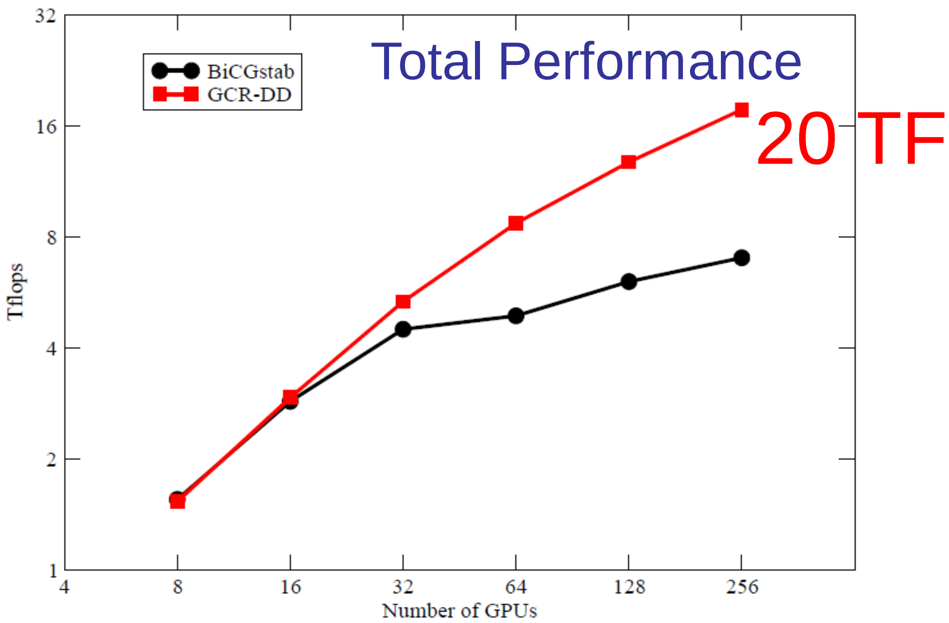
- Necessary software, services or infrastructure
  - LAPACK support & GPU library support
  - Possibly/probably heterogenous-system compiler support
  - Analysis: global file systems required for paging short term
  - Require parallel disk read/write for performance
  - Off-site network: no real-time comms, but requirements growing
  - Assumes capability & capacity co-located

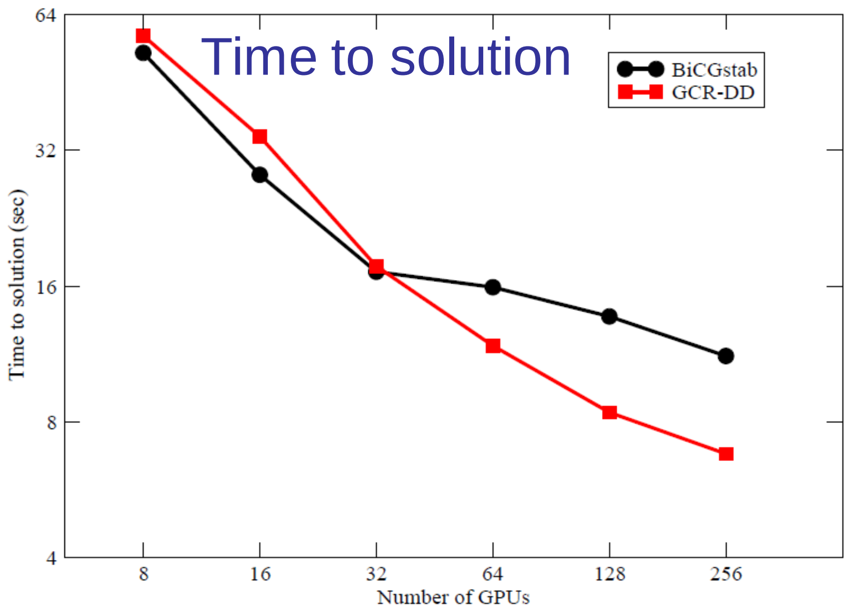
HPC Usage and methods for the next 3-5 years

- Anticipated limitations/obstacles/bottlenecks on 10K-1000K PE system.
  - Generically, limitation is usually comms/compute ratio
  - Obviously memory bandwidth an issue, but also latency

Strategy for new architectures

- Heterogeneous systems (GPU-s)?
  - Steep hierarchical memory subsystems
  - Suggests a domain decomposition strategy
  - Inverter: simple block-Jacobi, half-precision + GCR
    20 TF @ 256 GPUs (Edge @ LLNL)
  - JaguarPF (XT5) ~ 16K cores





Strategy for new architectures

Heterogeneous systems (GPU-s)?

- Possible path to exascale
- At 20 TF on 256 GPUs (128 boxes) – comparable to leadership
- BUT: only inverter (& half-precision)
- Great for capacity. Can we generalize??

Capability computing:

- Push domain decomposition into integrators
- Amdahl's law – formally small bits of code

Under SciDAC-3:

- Data-parallel domain specific extensions to data-parallel layer (QDP++)
- Portable/efficient on heterogeneous hardware – higher level code will "port"
- Wish collaboration with other institutes

Strategy for new architectures

Many-core with 10 – 100 cores/node??

- LQCD deeply involved: Columbia/BNL/Edinburgh co-designers of BG/Q
- Low cost of synchronizations
- Hybrid/threading model: already used
- Synchronizations not cheap and/or steep memory hierachy?
  - Domain decomposition techniques

Under SciDAC-3:

- Extend aggressive cache optimizations more into Data-Parallel